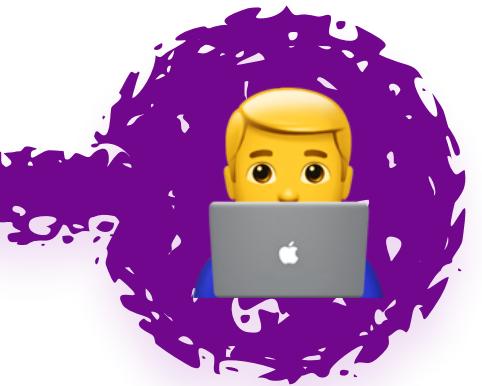


Aula 02

Estrutura de dados, lógica matemática e raciocínio lógico.

Academy



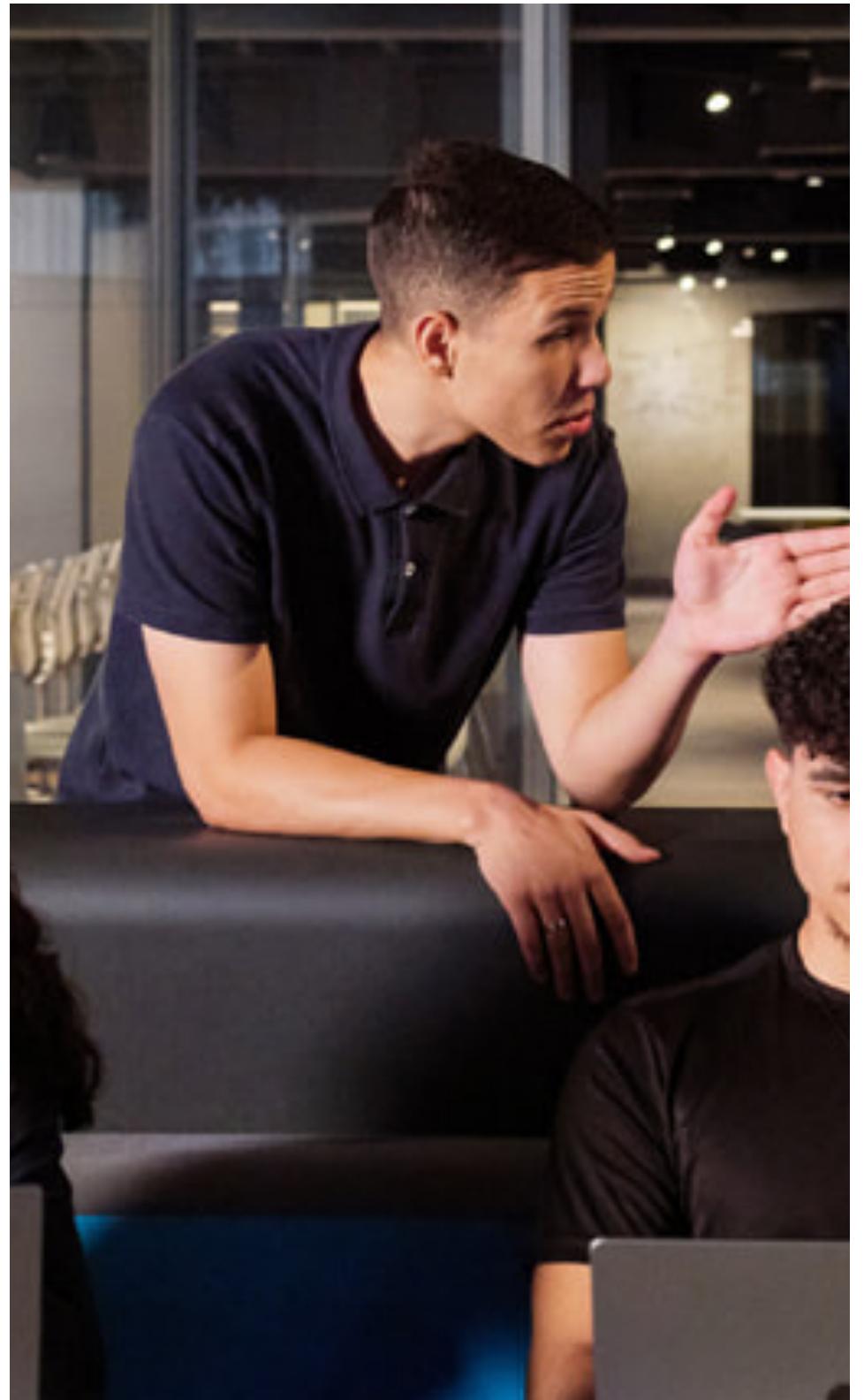
🧐 - Quem somos



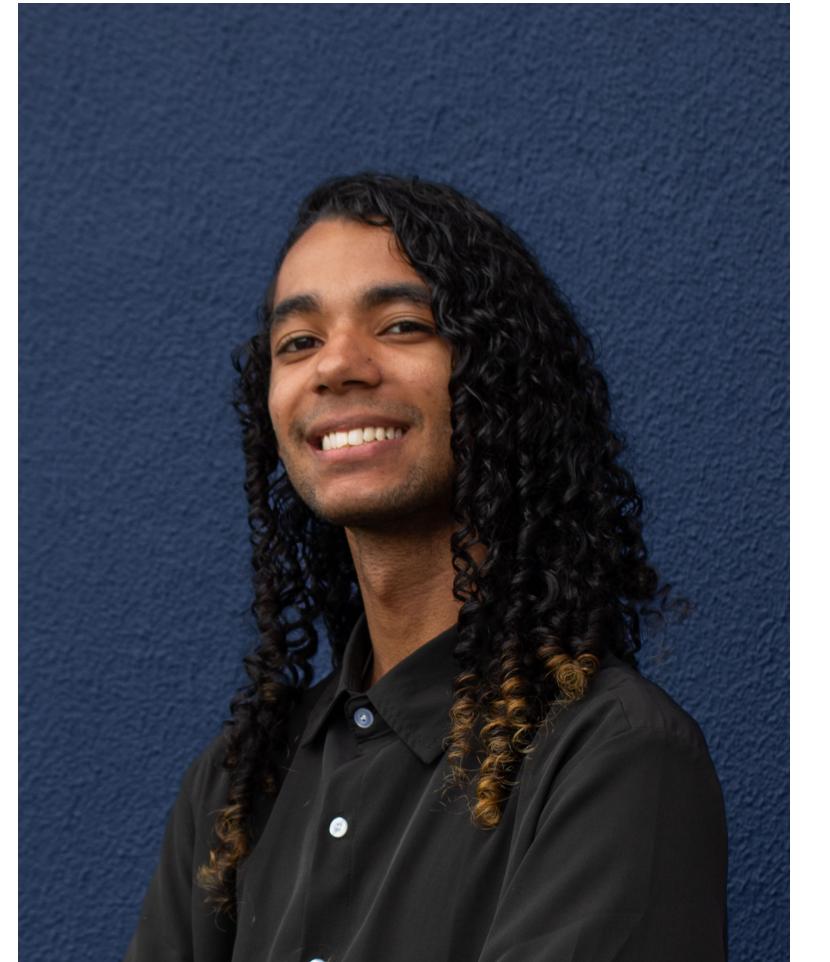
Leonardo Mesquita



Luca Gabriel



Kauã Miguel



João Victor

**Sintaxe em C, Ponteiros, Alocação de
Memória, Estrutura de dados, Lógica
Matemática, Programação Orientada
a Objetos.**

Nossa Prova - Conteúdos

Busca binária

Conceito

- Um algoritmo que busca um elemento em um array **ordenado**
- Segue o princípio de dividir para conquistar.

Busca binária

Conceito

```
int binarySearch(int vet[], int value, int left, int right) {  
    if (left > right) {  
        printf("Valor não encontrado! \n");  
        return -1;  
    }  
  
    int mid = (right + left) / 2;  
  
    if (value == vet[mid]) {  
        return vet[mid];  
    } else if (value > vet[mid]) {  
        return binarySearch(vet, value, mid + 1, right);  
    } else {  
        return binarySearch(vet, value, left, mid - 1);  
    }  
}
```

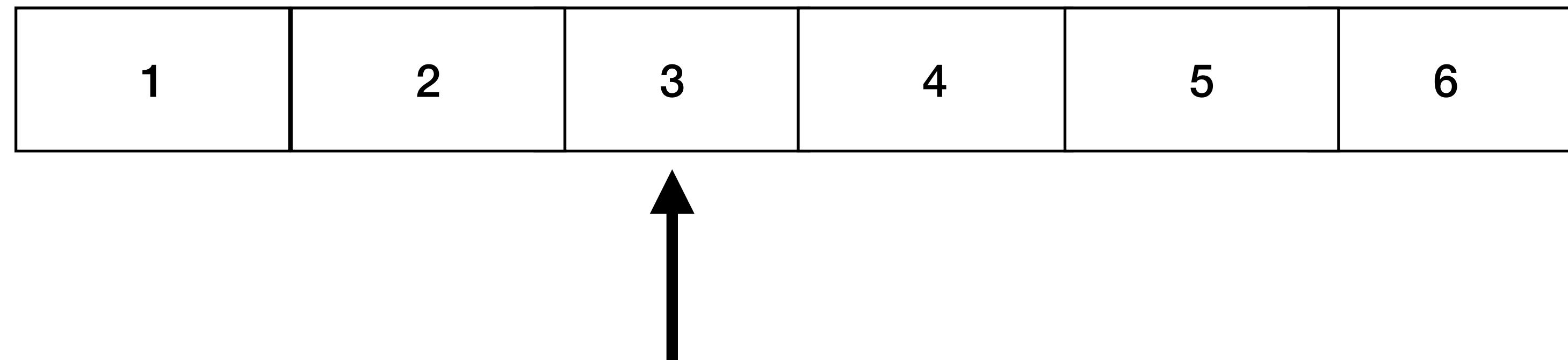
Busca binária

Conceito

```
int binarySearch(int vet[], int value, int left, int right) {  
    if (left > right) {  
        printf("Valor não encontrado! \n");  
        return -1;  
    }  
  
    int mid = (right + left) / 2;  
  
    if (value == vet[mid]) {  
        return vet[mid]; // Ponto de parada  
    } else if (value > vet[mid]) {  
        return binarySearch(vet, value, mid + 1, right);  
    } else {  
        return binarySearch(vet, value, left, mid - 1);  
    }  
}
```

Busca binária

Buscando o valor 1



Busca binária

Buscando o valor 1

1	2	3
---	---	---



4	5	6
---	---	---

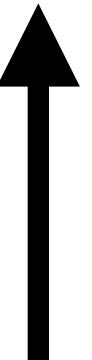
Busca binária

Buscando o valor 1

1

2	3
---	---

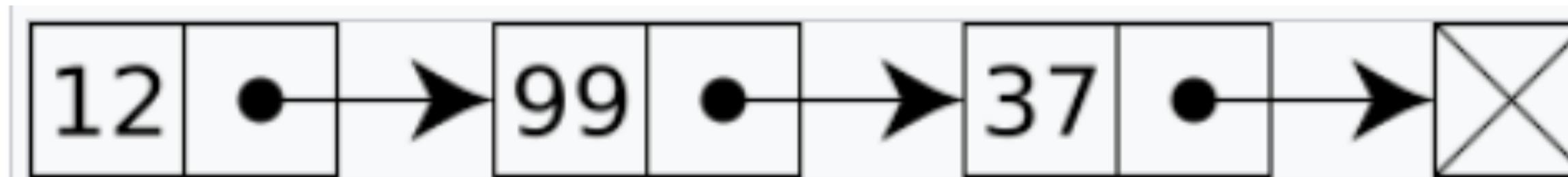
4	5	6
---	---	---



Listas encadeadas

Definição

- Uma estrutura de dados linear com uma sequência de objetos em locais diferentes da memória.
- Cada elemento é chamado de node, contendo um valor e uma referência pro próximo elemento



A linked list is a sequence of nodes that contain two fields: data (an integer value here as an example) and a link to the next node. The last node is linked to a terminator used to signify the end of the list.

Listas encadeadas

Definição

- O raciocínio de construir uma lista encadeada é o mesmo para construir pilhas e filas, mudando majoritariamente o método de acesso a dados.

Listas encadeadas

Operações principais

- Adicionar um node
- Remover um Node
- Criar a lista
- Pegar um item da lista

Listas encadeadas

Estrutura do node

```
typedef struct nodeitem{  
    int info;  
    struct nodeitem *next;  
}node;
```

```
typedef struct{  
    node *first;  
}linkedList;
```

Listas encadeadas

Criar a lista

```
linkedList *createList(){
    linkedList *list = malloc(sizeof(linkedList));
    list->first = NULL;
    return list;
}
```

Listas encadeadas

Verificar se está vazia

```
int isEmpty(linkedList *list){  
    if(list->first == NULL)  
        return 1;  
    return 0;  
}
```

Listas encadeadas

Criar um node da lista

```
node *createNode(int value){  
    node *node0fList = malloc(sizeof(node));  
    node0fList->info = value;  
    node0fList->next = NULL;  
    return node0fList;  
}
```

Listas encadeadas

Adicionar um elemento

```
void addItem(linkedList *list, int value, int position){

    node *aux = createNode(value);

    if(position == 1){
        //Insere na primeira posição;

        aux->next = list->first;
        list->first = aux;
    }else{
        //Insere em qualquer posição exceto a primeira;

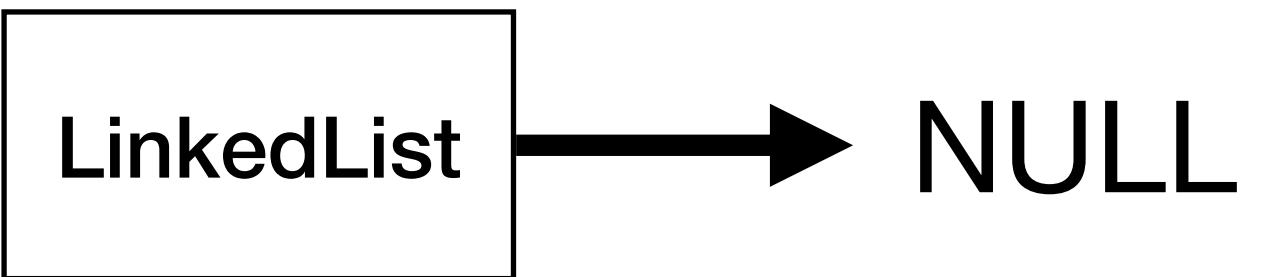
        node *ant = list->first;
        for(int i = 1; i < position - 1; ++i){
            ant = ant->next;
        }
        aux->next = ant->next;
        ant->next = aux;
    }

}
```

Listas encadeadas

Adicionar um elemento

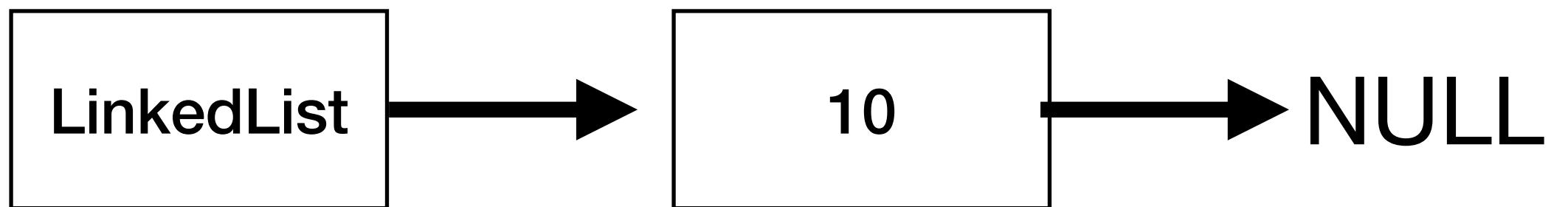
Cria nossa lista



Listas encadeadas

Adicionar um elemento

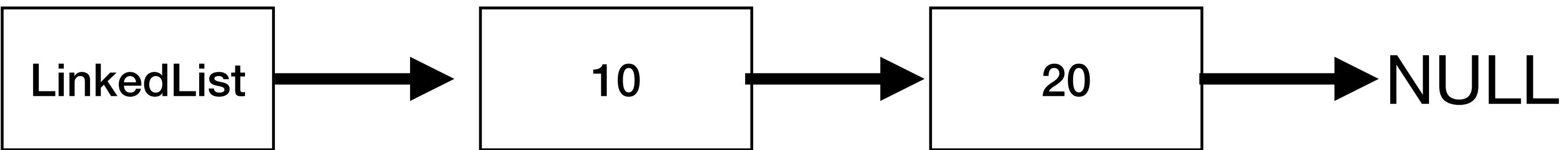
Insere o primeiro elemento



Listas encadeadas

Adicionar um elemento

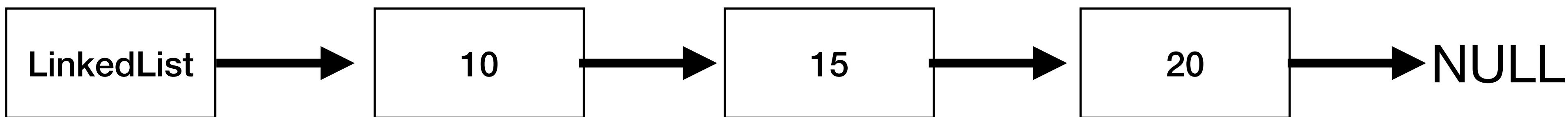
Insere o segundo elemento



Listas encadeadas

Adicionar um elemento

Insere um elemento em uma posição qualquer



Listas encadeadas

Mostrar os elementos da lista

```
void showList(linkedList *list){  
    if(isEmpty(list)){  
        return;  
    }else{  
        node *aux = list->first;  
        printf("List: ");  
  
        while(aux != NULL){  
            printf("[%d] ", aux->info);  
            aux = aux->next;  
        }  
        printf("\n");  
    }  
}
```

Listas encadeadas

Removendo elemento da lista

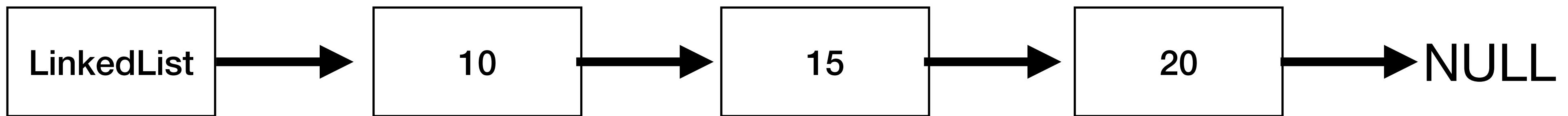
```
void removeItem(linkedList *list, int position){

    node *ant = list->first;

    if(position == 1){
        list->first = ant->next;
        free(ant);
    }else{
        for(int i = 1; i < position - 1; ++i){
            ant = ant->next;
        }
        node *aux = ant->next;
        ant->next = aux->next;
        free(aux);
    }
}
```

Listas encadeadas

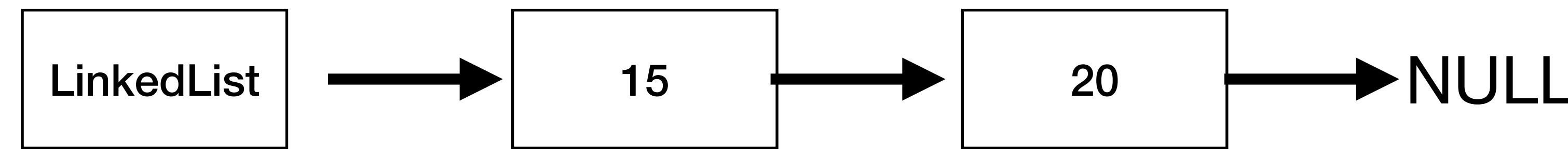
Removendo elemento da lista



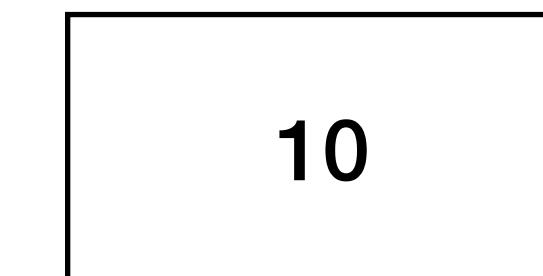
Listas encadeadas

Removendo elemento da lista

Removendo o primeiro elemento



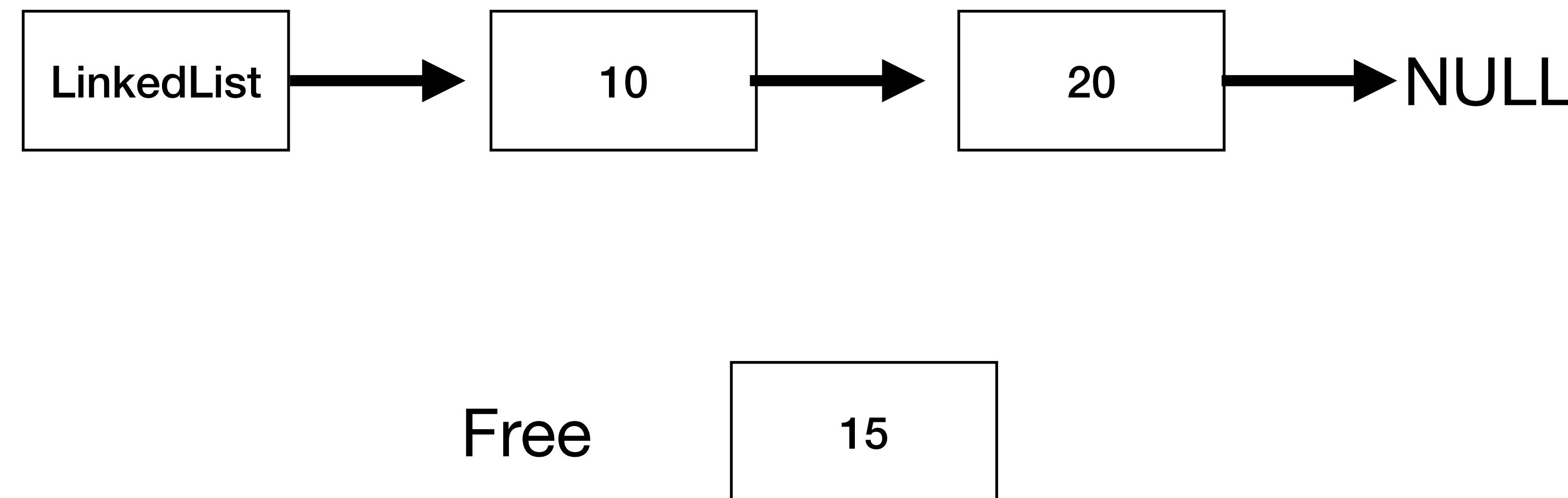
Free



Listas encadeadas

Removendo elemento da lista

Eliminando um elemento qualquer



Responda

Assinale a alternativa correta

- () Pilha é uma estrutura de dados LIFO A) V V F V
- () Filas são compostas por duas operações: Push e pop B) V F V F
- () Filas são FIFO C) F F V F
- () Pode-se inserir elementos de tipos diferentes em um array desde que a capacidade de armazenamento não sofra alterações D) F F V V

Responda

Assinale a alternativa correta

() Pilha é uma estrutura de dados LIFO

A) V V F V

() Filas são compostas por duas operações: Push e pop

B) V F V F 

() Filas são FIFO

C) F F V F

() Pode-se inserir elementos de tipos diferentes em um array desde que a capacidade de armazenamento não sofra alterações

D) F F V V

Responda

O que falta para inserir o nó no início e manter o duplo encadeamento?

```
typedef struct no{
    int valor;
    struct no *ant, prox;
} No;

No *insere (No *inicio, No *no){
    if(inicio == NULL){
        no->ant = no->prox = NULL;
        inicio = no;
    }else{
        // O que falta?
    }
    return inicio
}
```

- A no->ant = **NULL**;
 no->prox = inicio;
 inicio-> = **NULL**;
- B no->ant = **NULL**;
 No->prox = inicio;
 Início->ant = no;
 Início = no;
- C no->ant = **NULL**;
 Início->ant = no;
 Início = no;
- D no->ant = **NULL**;
 No->prox = inicio;
 Início->ant = **NULL**;
 Início = no;

Responda

O que falta para inserir o nó no início e manter o duplo encadeamento?

```
typedef struct no{
    int valor;
    struct no *ant, prox;
} No;
```

```
No *insere (No *inicio, No *no){
    if(inicio == NULL){
        no->ant = no->prox = NULL;
        inicio = no;
    }else{
        // O que falta?
    }
    return inicio
}
```

A

```
no->ant = NULL;
no->prox = inicio;
inicio-> = NULL;
```

B

```
no->ant = NULL;
No->prox = inicio;
Início->ant = no;
Início = no;
```

C

```
no->ant = NULL;
Início->ant = no;
Início = no;
```

D

```
no->ant = NULL;
No->prox = inicio;
Início->ant = NULL;
Início = no;
```

Responda

O que falta para inserir o nó no início e manter o duplo encadeamento?

```
No *insere (No *inicio, No *no){
    if(inicio == NULL){
        no->ant = no->prox = NULL;
        inicio = no;
    }else{
        no->ant = NULL; // Para ser o primeiro nao tem ninguem antes
        No->prox = inicio; // Meu atual aponta pra quem era o primeiro
        Inicio->ant = no; // O antigo primeiro aponta pro atual
        Inicio = no; // O inicio agora é o no criado
    }
    return inicio
}
```

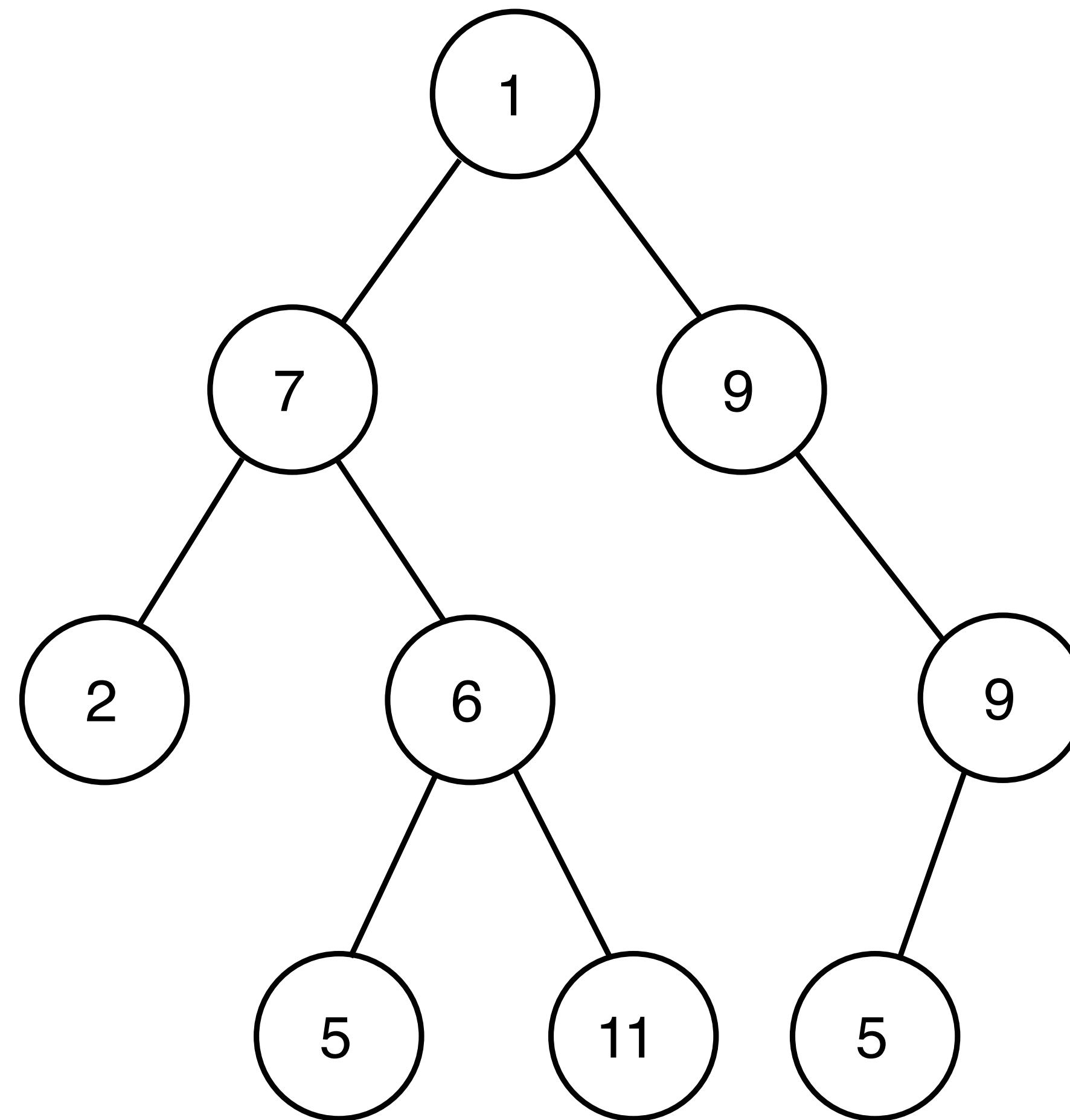
Árvore binária

Conceito

- Estrutura de dados hierárquica composta por nós que podem ter no máximo dois filhos.
- Cada filho é chamado de subárvore esquerda e direita.

Árvore binária

Conceito



Árvore binária

Percorso

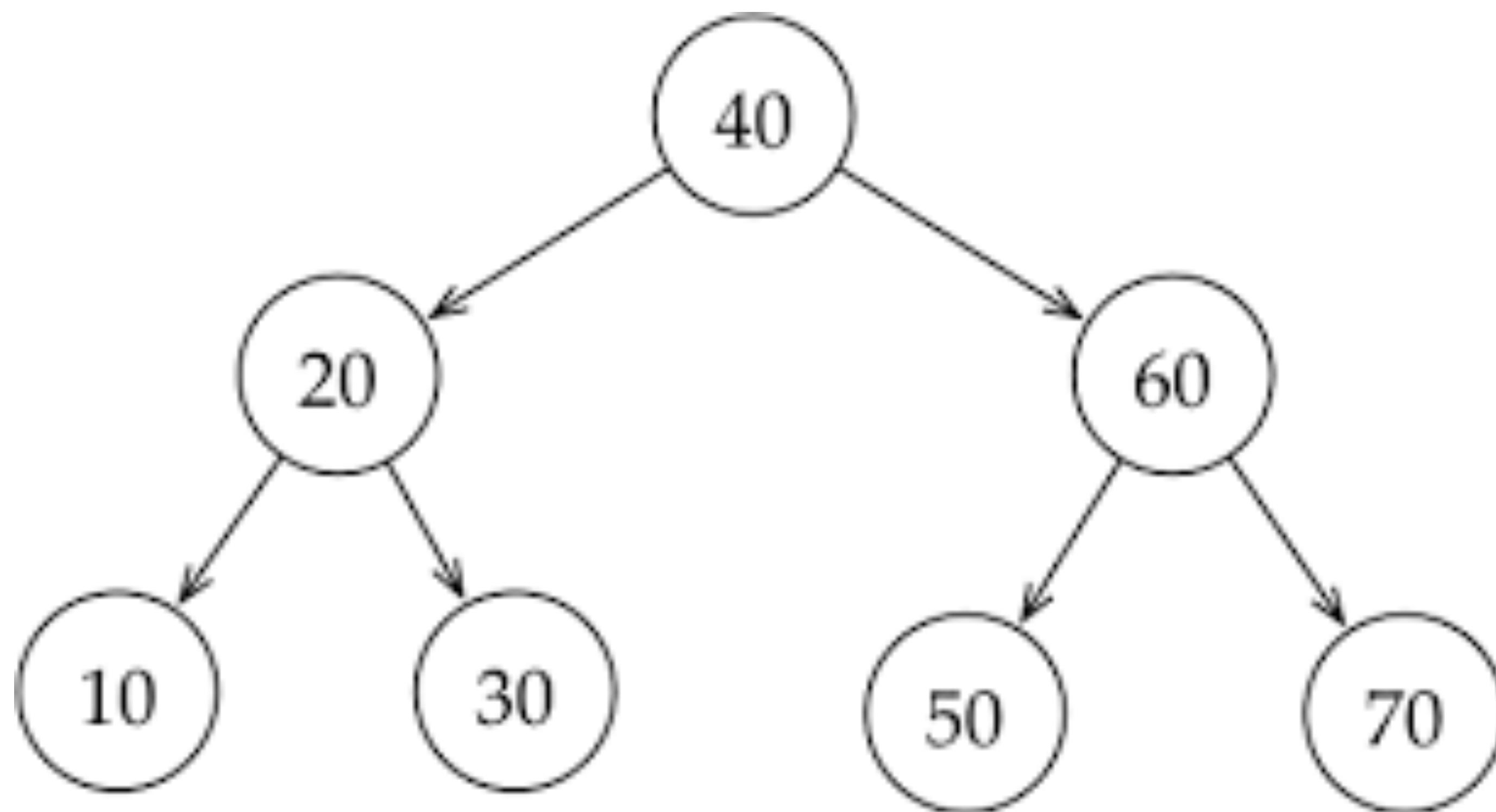
- Pré Ordem (Raiz, esquerda, direita)
- Em Ordem (Esquerda, raiz, direita)
- Pós Ordem (Esquerda, direita, raiz)

Árvore binária de busca

Conceito

- Uma árvore binária onde o elemento menor que a raiz está a esquerda e maior a direita.

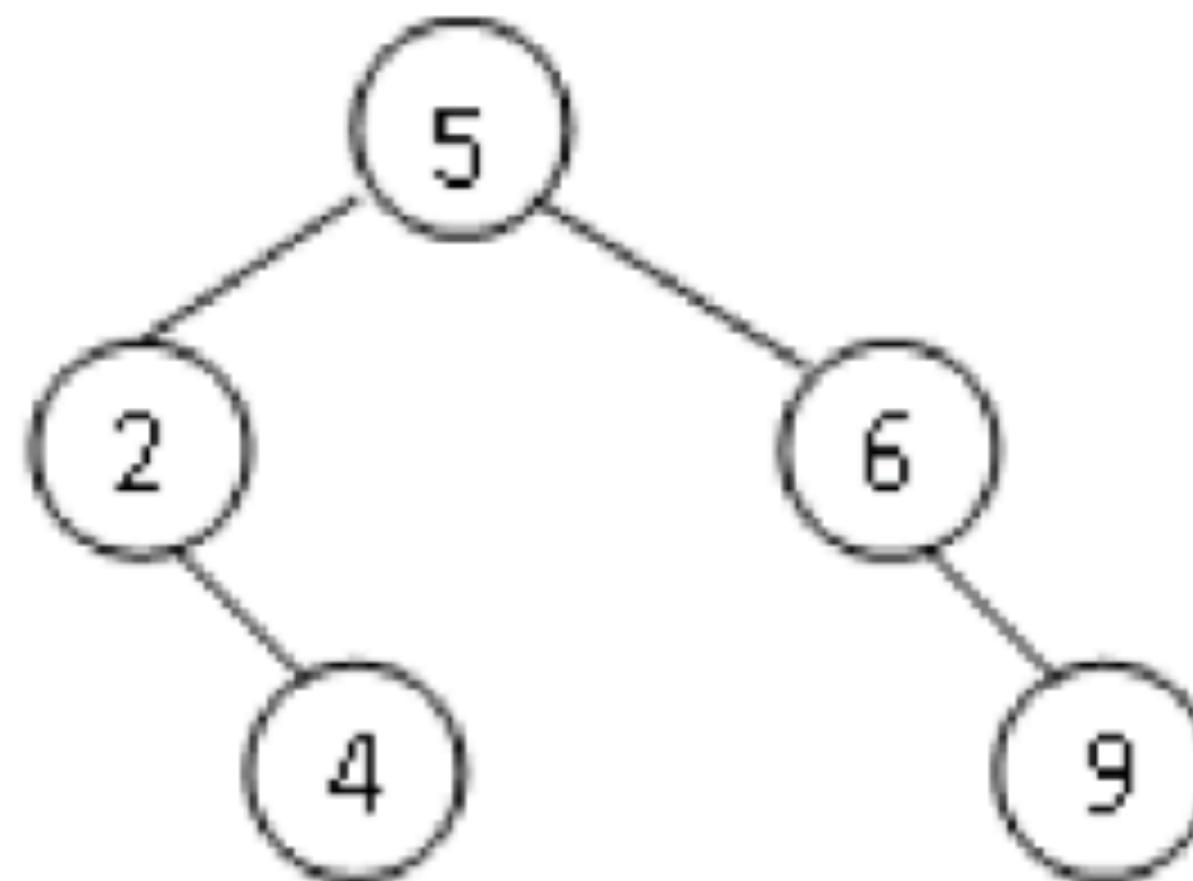
Árvore binária de busca



Árvore binária

Questão

13) Ao imprimir os valores contidos na árvore abaixo, percorrendo-a em pré-ordem, obtemos a seguinte sequência de valores:

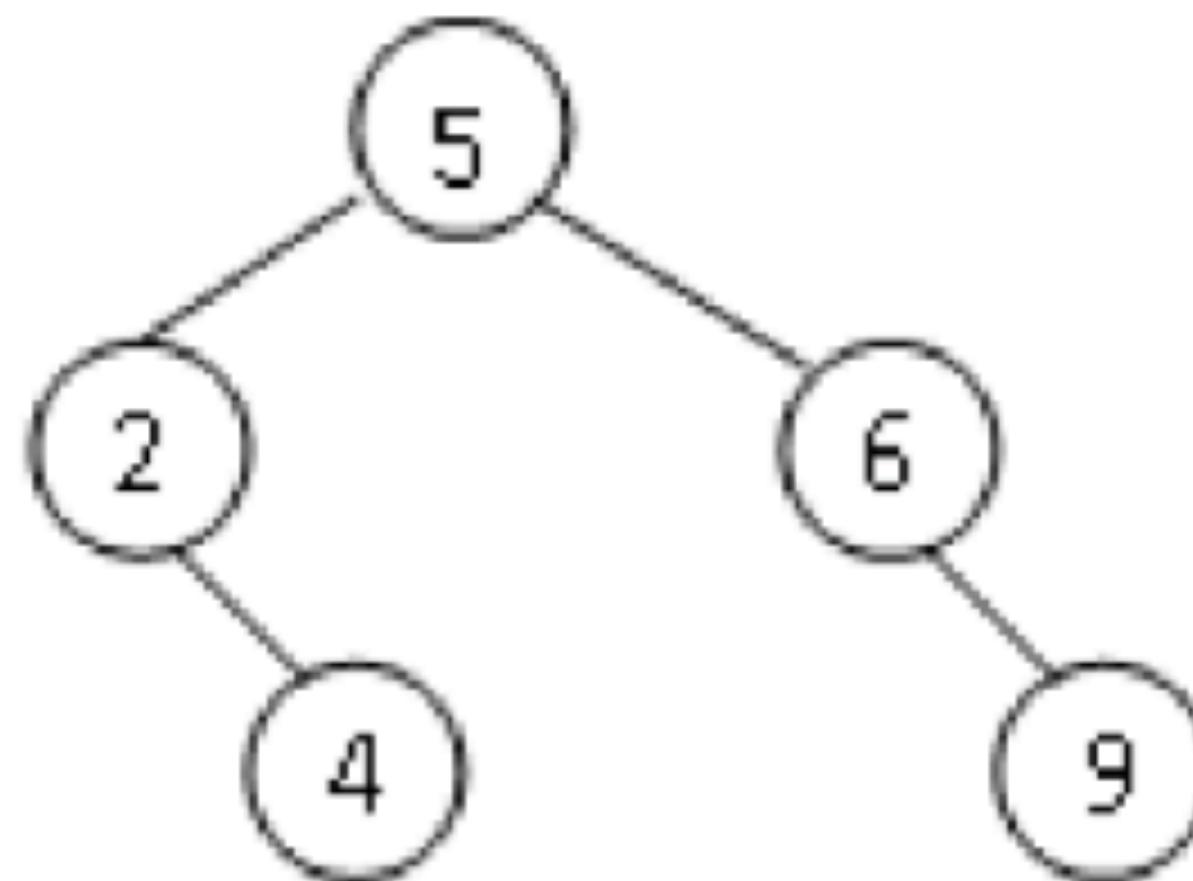


- a) 5, 2, 6, 4, 9
- b) 2, 4, 5, 6, 9
- c) 4, 2, 9, 6, 5
- d) 5, 2, 4, 6, 9
- e) 9, 6, 5, 4, 2

Árvore binária

Questão

13) Ao imprimir os valores contidos na árvore abaixo, percorrendo-a em pré-ordem, obtemos a seguinte sequência de valores:



- a) 5, 2, 6, 4, 9
- b) 2, 4, 5, 6, 9
- c) 4, 2, 9, 6, 5
- d) 5, 2, 4, 6, 9 ✓
- e) 9, 6, 5, 4, 2

Estrutura de dados

O que é essa função?

```
int funcao(int x, int v[], int e, int d){  
    int meio = (e + d) / 2;  
  
    if(v[meio] == x)  
        return meio;  
    if(e >= d)  
        return -1;  
    else  
        if(v[meio] < x)  
            return funcao(x, v, meio+1, d);  
        else  
            return funcao(x, v, e, meio-1);  
}
```

- A) Torre de hanoi
- B) Inserção na lista
- C) Fibonacci
- D) Dijkstra
- E) Busca binária

Estrutura de dados

O que é essa função?

```
int funcao(int x, int v[], int e, int d){  
    int meio = (e + d) / 2;  
  
    if(v[meio] == x)  
        return meio;  
    if(e >= d)  
        return -1;  
    else  
        if(v[meio] < x)  
            return funcao(x, v, meio+1, d);  
        else  
            return funcao(x, v, e, meio-1);  
}
```

- A) Torre de hanoi
- B) Inserção na lista
- C) Fibonacci
- D) Dijkstra
- E) Busca binária

Árvore binária

Questão

11) Uma versão especializada de uma lista encadeada em que os nós só podem ser inseridos no início da lista e excluídos do final da lista é denominada:

- a) Árvore ternária
- b) Pilha
- c) Árvore binária
- d) Fila
- e) Tabela hash

Árvore binária

Questão

11) Uma versão especializada de uma lista encadeada em que os nós só podem ser inseridos no início da lista e excluídos do final da lista é denominada:

- a) Árvore ternária
- b) Pilha
- c) Árvore binária
- d) Fila
- e) Tabela hash

Raciocínio Lógico

Questão

Para percorrer 250 KM um carro A gasta 60 Litros de gasolina. Supondo que exista o carro B que tem o consumo 4 vezes menor, a quantidade necessária para o mesmo percorrer 500 KM é?

Raciocínio Lógico

Questão

Se o carro B percorre o mesmo tanto com 4 vezes menos gasolina, então

$$\text{Carro B gasta } \frac{1}{4} * 60 = 15$$

Raciocínio Lógico

Questão

Se pra percorrer 250 KM ele usa 15 Litros, pra percorrer 500 KM ...

Raciocínio Lógico

Questão

Na porta da minha casa passam dois ônibus, A e B. Na casa ao lado moram dois irmãos. Um só diz verdade e outro só diz mentira. Ao indagar sobre qual ônibus tomar, um dos irmãos me disse : “Se o meu irmão estivesse aqui, mandaria você tomar o ônibus A”. Qual ônibus devo tomar?

Raciocínio Lógico

Solução

Na porta da minha casa passam dois ônibus, A e B. Na casa ao lado moram dois irmãos. Um só diz verdade e outro só diz mentira. Ao indagar sobre qual ônibus tomar, um dos irmãos me disse : “Se o meu irmão estivesse aqui, mandaria você tomar o ônibus A”. Qual ônibus devo tomar?

Se quem responde é o mentiroso, o irmão dele indicaria o ônibus correto mas o mentiroso diria o contrário.

Se quem responde é o verdadeiro, o irmão indicaria o errado, e ele falando a verdade diria que o irmão indicou o ônibus errado.

Resposta: Ônibus B

Raciocínio Lógico

Questão

Pedro e Paulo são irmãos. Pedro sempre mente e Paulo sempre diz a verdade. Uma pessoa fez uma pergunta à eles; um dos irmãos respondeu a primeira e outro a segunda.

- I) Qual é o seu nome, Pedro ou Paulo?
- II) Qual o nome do seu irmão, Pedro ou Paulo?

Qual a resposta obtida?

- A) Pedro e Pedro
- B) Pedro e Paulo
- C) Paulo e Pedro
- D) Paulo e Paulo
- E) Nada a afirmar

Raciocínio Lógico

Questão

Pedro e Paulo são irmãos. Pedro sempre mente e Paulo sempre diz a verdade. Uma pessoa fez uma pergunta à eles; um dos irmãos respondeu a primeira e outro a segunda.

- I) Qual é o seu nome, Pedro ou Paulo?
- II) Qual o nome do seu irmão, Pedro ou Paulo?

Qual a resposta obtida?

- A) Pedro e Pedro
- B) Pedro e Paulo
- C) Paulo e Pedro 
- D) Paulo e Paulo
- E) Nada a afirmar

Raciocínio Lógico

Questão

Pedro e Paulo são irmãos. Pedro sempre mente e Paulo sempre diz a verdade. Uma pessoa fez uma pergunta a eles; um dos irmãos respondeu a primeira e outro a segunda.

- I) Qual é o seu nome, Pedro ou Paulo?
- II) Qual o nome do seu irmão, Pedro ou Paulo?

Qual a resposta obtida?

A primeira pergunta Pedro responderia Paulo e Paulo responderia Paulo, pois o primeiro mente e o segundo diz a verdade.

A segunda pergunta Pedro responderia Pedro e Paulo responderia Pedro.

Lógica matemática

Conceito

- Proposição é toda sentença declarativa que pode ser classificada como verdadeiro ou falso.

Lógica matemática

Exemplo

- 2 é par
- Pi é irracional
- João comeu banana

Lógica matemática

Valores possíveis

- Verdadeiro ou Falso (V ou F)

Lógica matemática

Negação

- Dada uma proposição, definimos a negação de p , $\neg p$, como a proposição não p , que é o valor lógico contrário

Lógica matemática

Conektivos lógicos

- Operadores que unem duas proposições.

Lógica matemática

Exemplo

- Hoje está chovendo e ontem comi maçã
- Dois é par e pi é irracional

Lógica matemática

Conektivos fundamentais

- E (\wedge), Conjunção
- Ou , Disjunção
- Se \rightarrow então ,Condicional
- Se e somente se \leftrightarrow , Bicondicional
- Não \sim , negação

Lógica matemática

Conektivos fundamentais

- E (\wedge), Conjunção
- Ou , Disjunção
- Se \rightarrow então ,Condicional
- Se e somente se \leftrightarrow , Bicondicional
- Não \sim , negação

Lógica matemática

Se e somente se

p	q	$p \rightarrow q$
v	v	v
v	f	f
f	v	v
f	f	v

Obrigado!!!

Conversa fiado



Kauã Miguel



Kauã Miguel



Kauamiguel_